

# Tarea complementaria 1 Rutas absolutas y relativas

## Objetivo

Comprender la diferencia entre **ruta absoluta** y **ruta relativa** al trabajar con ficheros en un sistema operativo, especialmente en el contexto de un proyecto de programación.

## 1) Definiciones

### ◆ Ruta absoluta

Una **ruta absoluta** es aquella que indica la **ubicación completa** de un fichero o directorio **desde la raíz del sistema**.

Siempre es **única**, porque describe el camino exacto hasta el recurso.

 *Definición corta:*

“Ruta que parte desde el directorio raíz del sistema y define toda la jerarquía hasta el fichero.”

Ejemplos:

- **Windows:**

C:\Usuarios\David\Documentos\proyecto\miFichero.txt

- **Linux/macOS:**

/home/david/Documentos/proyecto/miFichero.txt

 Las rutas absolutas **no dependen del programa** ni de la carpeta de ejecución, sino de la estructura fija del sistema operativo.

### ◆ Ruta relativa

Una **ruta relativa** indica la **ubicación de un fichero en relación con el directorio actual** del programa (también llamado *working directory*).

No incluye la raíz del sistema, sino que parte del punto donde se ejecuta la aplicación.

 *Definición corta:*

“Ruta que se construye en relación al directorio de trabajo del proyecto.”

Ejemplos:

- src/tarea/obligatoria/pkg1/clase/file/java/miDirectorio/miFichero.txt
- ./AD/Ejercicios/nuevoDirectorio/fichero\_de\_texto2.txt  
( ./ indica el directorio actual)

 Las rutas relativas son **portables** y funcionan en distintos sistemas operativos sin necesidad de modificarlas.

## 2) Ejemplo real en mi sistema

### ◆ En Linux

Supongamos que tengo un archivo de texto en mi carpeta personal:

```
/home/endurance/Escritorio/TAREA_OBLIGATORIA_1/miFichero.txt
```

- Esa es la **ruta absoluta** (empieza desde `/home` ).
- Si mi programa se ejecuta desde la carpeta del proyecto, puedo acceder al mismo archivo con una **ruta relativa** como:

```
../TAREA_OBLIGATORIA_1/miFichero.txt
```

(`..` indica “subir un nivel” desde el directorio actual)

## 3) Preguntas y respuestas

### a) ¿Qué ventaja tiene usar rutas relativas en un proyecto de programación?

Las **rutas relativas** permiten que el programa sea **portable** y pueda ejecutarse en diferentes ordenadores o sistemas operativos **sin modificar las rutas de los ficheros**.

Esto es fundamental cuando el proyecto se comparte, se compila en otro entorno o se sube a un **repositorio** (por ejemplo, GitHub o un campus virtual).

 En resumen: las rutas relativas facilitan la portabilidad, la organización del código y el trabajo en equipo.

### b) ¿En qué casos es más recomendable usar una ruta absoluta?

Las rutas absolutas se recomiendan cuando:

- Se trabaja con **recursos del sistema** que siempre estarán en la misma ubicación (por ejemplo, `/etc/hosts` en Linux o `C:\Windows\System32` en Windows).
- Se necesita **acceder a ficheros fuera del proyecto**, como logs del sistema, directorios de configuración global o bases de datos instaladas en ubicaciones fijas.
- El programa se ejecuta como servicio o script del sistema operativo, donde la ruta de trabajo puede variar.

 En resumen: las rutas absolutas se usan para ubicaciones **fijas y controladas**, mientras que las relativas son mejores para **proyectos portables y colaborativos**.